Homework 4

Jenian Tai (kt2694, kt2694@columbia.edu)

About The Data

We will be working with a simulated data set related to social media sites. The data are stored in several files:

Profiles.csv: Information about the users with some fields from their profiles.

Connections.csv: Information about which users are connected to other users.

Registrations.csv: Information about history of the user's account registrations (logins) over time.

Header The first row of the data set includes the column names, and each subsequent row includes one observation of values. Here is a selection of 20 lines from each data file:

Show 10 ▾ entries   Search:

	id	density	age_group	gender	has_profile_photo	num_photos	date_created
1	0009g7fE	Suburban	18-24	F	1	41	2012-04-17
2	003fp4C4	Rural	35-44	M	1	10	2015-08-14
3	008Xfcs6	Rural	45-54	M	1	834	2016-03-17
4	00lKIVvN	Urban	18-24	F	1	927	2013-09-30
5	01sqmyAe	Suburban	55-64	M	1	1096	2013-01-14
6	02J5xPk5	Urban	18-24	M	1	78	2016-08-23
7	02Kl5Mtk	Rural	18-24	M	1	515	2016-08-13
8	02sMiAYv	Suburban	45-54	F	0	1184	2015-05-15
9	02uJHCPG	Suburban	45-54	M	1	468	2015-11-17
10	0317C7rA	Suburban	25-34	F	1	400	2016-08-11

Showing 1 to 10 of 20 entries   Previous 1 2 Next

Show 10 ▾ entries   Search:

	id	connection_id
1	0009g7fE	0LsKhl8z
2	0009g7fE	0MukfMFq
3	0009g7fE	0RaKGhQ3
4	0009g7fE	0RhatvnN
5	0009g7fE	0hJwTkNt
6	0009g7fE	1SI6BX3W

	id	connection_id
7	0009g7fE	1kmYuJTc
8	0009g7fE	27PNx2Kz
9	0009g7fE	2IN0UsqQ
10	0009g7fE	2Ogw7nOx

Showing 1 to 10 of 20 entries

Previous 1 2 Next

Show 10 ▾ entries

Search:

	id	registration.time	original.registration.time
1	0009g7fE	2012-04-17T01:16:36Z	2012-04-17T01:16:36.638355Z
2	0009g7fE	2012-04-17T17:12:50Z	2012-04-17T17:12:50.185053Z
3	0009g7fE	2012-04-17T19:50:52Z	2012-04-17T19:50:52.626801Z
4	0009g7fE	2012-04-18T02:31:24Z	2012-04-18T02:31:24.478748Z
5	0009g7fE	2012-04-18T02:58:57Z	2012-04-18T02:58:57.193260Z
6	0009g7fE	2012-04-18T11:28:34Z	2012-04-18T11:28:34.407250Z
7	0009g7fE	2012-04-18T14:56:02Z	2012-04-18T14:56:02.268198Z
8	0009g7fE	2012-04-19T08:04:03Z	2012-04-19T08:04:03.395474Z
9	0009g7fE	2012-04-19T14:24:15Z	2012-04-19T14:24:15.157158Z
10	0009g7fE	2012-04-20T04:48:46Z	2012-04-20T04:48:46.500991Z

Showing 1 to 10 of 20 entries

Previous 1 2 Next

Here is a brief description of each variable across the three files:

Profiles Variables:

- **id**: A unique identifying string for each user.
- **density**: The type of area the user lives in, with categories of Urban, Suburban, and Rural areas.
- **gender**: female (F) or male (M).
- **has_profile_photo**: 1 if yes, 0 if no.
- **num_photos**: This is the number of photos the user has uploaded to the site.
- **date_created**: This is the date that the user first joined the site.

Connections Variables:

- **id**: A unique identifying string for each user.
- **connection_id**: This is the identifier of another user that the user listed under **id** is connected to.

This site chooses to use one-way connections. A user can connect to a second user's profile without requiring that the second user reciprocally connect to the first one. So, for any row in the Connections data, the user labeled with **id** is following the user labeled with **connection_id**. In some cases, pairs of users are mutually following each other, but this is by no means required. For mutual connections, the users will be coupled in two different rows in the two possible orders. Each connection for a single user is recorded in a separate row.

Registrations Variables:

- **id**: A unique identifying string for each user.
- **registration.time**: This is the date and time that a user registered by logging in to the site. Each registration for a user is recorded in a separate row.

Question 1: Classifying Connections

How often do users mutually follow each other, and how often are the connections one-way? We want to investigate this. For the investigation, we'll say that a two-way connection requires two one-way connections (two rows of data) but only counts once. Therefore, the number of overall connections (total one-way plus total two-way) will be less than the overall number of rows of data in the Connections file. With this in mind, answer these questions.

What percentage of all connections are one-way connections, and what percentage of all connections are two-way connections?

```
# functions unique and duplicated: one-way includes the duplicated once, and two-way includes the dup
flip_id <- as.data.table(t(apply(connections, 1, sort)))
one_way <- unique(flip_id)
two_way <- flip_id[duplicated(flip_id)]
paste("One-way: ", round((nrow(one_way)-nrow(two_way))/nrow(connections)*100,2),'%')
```

```
[1] "One-way:  32.3 %"
```

```
paste("Two-way: ", round(nrow(two_way)/nrow(connections)*100,2),'%')
```

```
[1] "Two-way:  33.85 %"
```

Question 2: Recommending Connections

Which connections should we recommend to the user with id CLKcSSSC? One way is to find the unconnected users who are connected to users that user CLKcSSSC is also connected to. Create a table of all the users who satisfy all of the following criteria:

- have at least 30 connections in common with user CLKcSSSC's connections, and
- are not already connected with user CLKcSSSC.

The list should show the ids of the recommended users and the number of common connections they have with user CLKcSSSC. Order the list in decreasing order of mutual connections. Make sure not to include CLKcSSSC on the list of recommendations!

```
the_connections <- connections[get(id.name) == the.id]
unconnected <- connections[!(connections$id %in% the_connections$connection_id)]
unconnected <- unconnected[,length(intersect(the_connections$connection_id, get(connection.id.name))
unconnected <- unconnected[V1 >=min.common.connections & get(id.name) != the.id][order(-V1)]
unconnected
```

```
          id V1
 1: KaBFl3mi 34
 2: be4WGFRz 33
 3: 261Vpi2U 31
 4: 7KgwsKvw 31
 5: YYEIYHrg 31
 6: gfCgPhAM 31
 7: 8XqINMTF 30
 8: 8fOTxvTy 30
 9: E2HSTCj6 30
10: FQTUyrmn 30
11: L7gdqIhN 30
12: NuNJRLuy 30
```

Question 3: Influential Connections

In social networks, some users are considered **influential**. They tend to have more connections, and their content can be widely viewed and shared. For our purposes, we will define the **influential users** as those who:

- Have at least 200 photos, and
- Have at least 150 connections.

Among all users (both influential and not so influential), how many users are connected to at least 250 **influential** users?

```
manyphoto <- profiles[num_photos>= min.photos.q3]
connection_sum <- connections[,.(follow = .N), by= id.name]
influencer <- merge(manyphoto, connection_sum, by=id.name)[follow>=min.connection.connections.q3]
connect_influencer <- connections[,length(intersect(influencer$id, get(connection.id.name))), by= id
nrow(connect_influencer)
```

```
[1] 2380
```

Question 4: Early Utilizers

Starting from the time when the account for each user was created, what percentage of all users logged in at least 35 times during the first 7? Round your answer to 1 decimal point, e.g. 84.2%.

Hints: Within the **lubridate** library, you can use the function **days** to add a specified number of days to the registration times. The first week ends before (less than) the user's first registration time plus 7 days. The registration that occurred when the account was created counts toward the overall total for this period.

```
registrations <- registrations[, first.registration.time := min(registration.time), by = id.name]
registrations <- registrations[, day_diff := difftime(registrations$registration.time, registrations
early_user <- registrations[, sum(day_diff<first.x.days), by=id.name][V1>=35]
paste(round(nrow(early_user)/length(unique(registrations$id))*100,1),"%")
```

```
[1] "29.8 %"
```

Question 5: Imbalanced Connections

What percentage of users have at least 100 more followers than the number of users that they are following? Round the answer to 1 decimal place, e.g. 84.2%.

```
being_follow <- connections[,.(being.follow = .N), by=connection.id.name]
names(being_follow)[1] <- "id"
follow_being_follow <- merge(connection_sum, being_follow, by=id.name)
paste(round.numerics(nrow(follow_being_follow[being.follow-follow>=100])/nrow(profiles)*100,1),'%')
```

```
[1] "32.8 %"
```

Question 6: Active Users

What percentage of unique users in the sample were active (with at least 1 registration) between 00:00:00 of January 1st, 2017 and 23:59:59 on January 7th, 2017? Round the percentage to 1 decimal place, e.g. 84.2%

Hint: For any given date in character format (e.g. “1999-07-01”), you can calculate a date in the future with the **as.Date** function: as.Date(“1999-07-01”) + 3 would result in “1999-07-04”.

```
start <- as.POSIXct(strptime("17-01-01", "%y-%m-%d"))
end <- as.POSIXct(strptime("17-01-08", "%y-%m-%d"))
active <- registrations[registration.time>= start & registration.time< end]
paste(round(uniqueN(active$id)/length(unique(registrations$id))*100,1),"%")
```

```
[1] "8.6 %"
```

Question 7: Burning the Midnight Oil

Across all days, what percentage of all registrations occur between the hours of 00:00:00 and 05:59:59, inclusive of both endpoints? Round your answer to 1 decimal place, e.g. 84.2%. **Hint:** Use the hour() function to classify the time of day.

```
midnight <- registrations[, log_hours := hour(registration.time)]
paste(round(midnight[, sum(log_hours >= 0 & log_hours < 6)]/nrow(midnight)*100,1),"%")
```

```
[1] "25 %"
```

Question 8: Retention Rates

What percentage of users were retained at 183 days (half a year)? To answer this question, we will use a 7 day window. Any user who had at least one registration in the period of time that was at least 183 days and less than 190 days from their first registration would be considered retained. Round your answer to 1 decimal place, e.g. 84.2%.

Note: The evaluation window would begin at exactly 183 days after the first registration. This period lasts for 7 days. This window would include the left end-point but not the right end-point. The registration times are listed in the data set rounded to the nearest second. If the user had at least 1 registration during this window, the user would be considered retained at 183 days (approximately 6 months).

Hint: You may use the **days()** function to add time to a user’s initial registration time.

```
retained_user <- registrations[,.(num_log = sum(day_diff>=183 & day_diff<190)), by = id.name][num_lo
paste(round(nrow(retained_user)/length(unique(registrations$id))*100,1),"%")
```

```
[1] "30.2 %"
```

Question 9: False Positive Rates

In the previous question, we estimated the rate of retention at 6 months using a 7-day window for evaluation. What is the rate of false positives for the 7-day window? In other words, what percentage of users who were considered not retained at 6 months using a 7-day window later had a registration? Round the results to 2 decimal places, e.g. 84.23%.

```
# users have registration after 190 days (included)
retained_user_miss <- registrations[,sum(day_diff>=190), by = id.name][V1>=1]
names(retained_user_miss)[2] <- "num_log"

# remove the overlapped users who both have registrations in [183, 190) and in [190, )
with_overlaps <- rbind(retained_user, retained_user_miss, fill=TRUE)
retained_miss_final <- with_overlaps[, .N, by=id.name][N==1]

# calculate the false positve rate - users have registrations after 190 days/users we thought they a
paste(round(nrow(retained_miss_final)/(length(unique(registrations$id))-nrow(retained_user))*100,1),
```

```
[1] "0.4 %"
```

Question 10: Modeling Retention

Build a logistic regression model for retention at 6 months. Classify users as retained at 6 months if they have any account registrations at times at least 183 days after their account was created. Include the following variables:

- density
- age_group
- gender
- num_photos (categories: 0-24, 25-49, 50-99, 100-249, 250-499, 500+) (current status)
- average daily registrations in the first week. (To simplify matters, let this be the total number of registrations in the first week divided by 7, regardless of whether the user's retention truly lasted 7 days or not.)
- number of connections the user currently has
- number of users currently connected to this user

Display the odds ratios, confidence intervals for the odds ratios, and p-values for the coefficients, rounded to 3 digits. Then briefly comment on the results.

```
# generate the dependent variable
retained_user[, if_retained := TRUE]

# generate independent variable - num_photos groups
cuts.photos <- c(25, 50, 100, 250, 500)
photo.group.name <- "num.photo.group"
profiles[, eval(photo.group.name) := cut2(x = num_photos, cuts = cuts.photos)]

# generate the dependent variable - average_daily registration by id
```

```
registrations_new <- registrations[, first_week_avg_log := sum(day_diff<=7)/7, by = id.name][,c(1,7)
registrations_new <- unique(registrations_new)

# merge the table with relevant variables
model_dat <- merge(merge(merge(profiles, registrations_new, by=id.name),
                         follow_being_follow, by=id.name, all.x = TRUE, all.y = FALSE),
                   retained_user, by=id.name, all.x = TRUE, all.y = FALSE)
model_dat[is.na(if_retained)==TRUE, if_retained := FALSE]

# construct the model
input.names <- c("density", "age_group","gender", "num.photo.group",
                 "first_week_avg_log", "follow", "being.follow")
model.q10 <- fit.model(dt = model_dat, outcome.name = "if_retained", input.names=input.names)
datatable(model.q10)
```

Show 10 ▾ entries Search:

	rn	Estimate	Std. Error	z value	Pr(>|z|)	Odds.Ratio	OR.Lower.95	OR.Upper.9
1	(Intercept)	-4.449	0.204	-21.799	0	0.012	0.008	0.0
2	densitySuburban	-0.609	0.101	-6.047	0	0.544	0.447	0.6
3	densityUrban	-1.305	0.102	-12.843	0	0.271	0.222	0.3
4	age_group25-34	-0.129	0.131	-0.984	0.325	0.879	0.68	1.1
5	age_group35-44	-0.359	0.123	-2.915	0.004	0.698	0.548	0.8
6	age_group45-54	-0.751	0.12	-6.24	0	0.472	0.373	0.5
7	age_group55-64	-1.177	0.138	-8.509	0	0.308	0.235	0.4
8	age_group65+	-0.45	0.151	-2.979	0.003	0.637	0.474	0.8
9	genderM	-0.339	0.075	-4.496	0	0.713	0.615	0.8
10	num.photo.group[25, 50)	-0.018	0.154	-0.114	0.909	0.983	0.727	1.3

Showing 1 to 10 of 17 entries

Previous 1 2 Next

If seems that many positive features (odds.ratio > 1) are not statistically significant. Among them, only “first_week_avg_log” and “follow” have p-value < 0.05. For negative features, we have more significant variables, and only a few such as “num.photo.group” and “age_group25-34” are not statistically significant.